公钥加密算法

2020年3月31日 14:28

1. 对称密码算法的问题
   a. 密钥管理成为用户的负担
   b. 对拥有庞大用户数量的网络的通信空间提出了很高的要求
   c. 密钥不能被及时更换以保证信息的保密性
   d. 数据的完整性得不到保证
   e. 无法保证接收到的消息来自于声明的发送者

-
- 非对称密码算法

1. 对称密码系统的缺点
   a. 密钥难以传输：对称密码体制需要通信前先通过一个安全信道交换密钥
   b. 密钥管理成为用户负担：n个人两两通信需要n(n-1)/2个不同密钥
   c. 不提供法律证据
   d. 缺乏自动检测密钥泄密的能力
2. 数论基础
   a. 基本概念
      i. 整除 $a|b \Leftrightarrow b=ak$
      ii. 素数（质数）
      iii. 最大公约数 gcd (a,b)
      iv. 同余 $a \equiv b \ mod \ n$
      v. 除法
   b. 欧几里得算法 gcd (a,b)=gcd $(b, a \ mod \ b)$
      i. $A \leftarrow a, \ B \leftarrow b$
      ii. 若B=0，则返回A= gcd (a,b)
      iii. $R \equiv A \ mod \ B$
      iv. $A \leftarrow B$
      v. $B \leftarrow R$
      vi. 转到ii.
   c. 乘法逆元
      i. 如果gcd (a,b)=1，那么存在a^(−1)，使$a \times a$^(−1)$\equiv 1 \ mod \ b$，称a^(−1) 为a模b的乘法逆元
      ii. 扩展的欧几里得算法 (递归)
      iii. gcd (a,b)=gcd $(b, a \ mod \ b)=d$
      iv. 假设已经求出了gcd $(b, a \ mod \ b)$的线性组合bx^′+$(a \ mod \ b)y'$，则

v. gcd $(a,b)=d=bx^{\prime}+(a \bmod b)y^{\prime}$

vi. $=bx^{\prime}+(a-[a/b]b)\ y^{\prime}=ay^{\prime}+b(x^{\prime}-[a/b]\ y^{\prime})$

vii. 所以$x=y^{\prime}$, $y=x^{\prime}-[a/b]\ y^{\prime}$

d. 费尔马小定理：如果p是素数，且a不是p的倍数，则$a^{(p-1)}\equiv 1 \bmod p$

e. 欧拉函数和欧拉定理

i. 欧拉函数$\phi(n)$：小于n且与n互素的正整数的个数

ii. 欧拉定理：对于任意互素的整数a和n，有$a^{(\phi(n))}\equiv 1 \bmod n$

f. 离散对数

i. 本原根：

1) $a^{m}\equiv 1 \bmod n$成立的最小正幂m，称为a的阶

2) 使$a^{m}\equiv 1 \bmod n$成立的最小正幂m满足$m=\phi(n)$，称a是n的本原根

ii. 离散对数：某素数p，有本原根a，且$X_1=a^{1} \bmod p$, $X_2=a^{2} \bmod p$, $\cdots$,$X_{(p-1)}=a^{(p-1)} \bmod p$, 那么有$X_1\neq X_2\neq\cdots\neq X_{(p-1)}$。令$S=\{X_1,X_2,\cdots,X_{(p-1)}\}$, $T=\{1,2,\cdots,p-1\}$，显然$S=T$。对任意正整数b，可以使得$b\equiv r \bmod p\equiv a^{i} \bmod p$，其中$0\leq r,i\leq p-1$。称指数i为模p的b的指标，或称离散对数，记作$[\![ind]\!]_{(a,p)}(b)$。

iii. 求离散对数是一个困难问题

3. 公钥密码算法

a. 和对称密码算法最大的不同在于通信双方使用不同的密钥

b. 除了用于加密，当用户使用自己的私钥对消息进行加密，而使用公钥解密时，非对称密码算法还可以实现签名的功能，保证了数据的完整性

c. 为了安全，需要保证私钥不易从明文或密文推出，且不易从公钥推出

d. 1976年，由 Diffie和Hellman提出公钥密码体制的思想；

e. 1977年，由Rivest、Shamir和Adleman发明了著名的RSA密码体制；其安全性基于分解大整数的困难性；

f. ElGamal密码体制：安全性基于离散对数；

4. 限门单向函数

a. 单向函数（One-way Function）：一个函数容易计算但难于求逆；

b. 限门单向函数（Trapdoor one-way function）：如果它是一个单向函数，并在具有特定限门的知识后容易求逆；

5. 公钥密码体制基本构成

a. 明文M：算法的输入

b. 密文C：算法的输出

c. 公钥K_e和私钥K_d：算法的输入，公钥公开，私钥保密

d. 加密、解密算法

6. 公钥加密体制的特点

a. 加密和解密能力分开

b. 多个用户加密的消息只能由一个用户解读，可用于公共网络中实现保密通信

c. 用私钥加密的消息可以用对应的公钥解密，所以由一个用户加密消息而使多

个用户可以解读，可用于认证系统中对消息进行数字签字

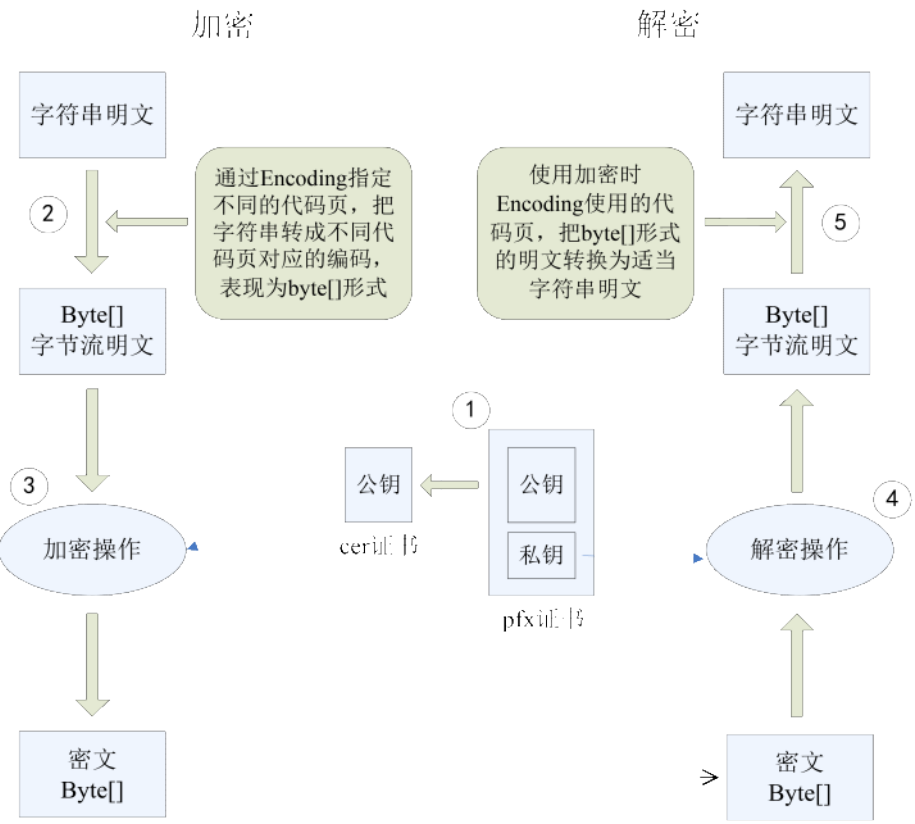
d. 无需事先分配密钥
e. 密钥持有量大大减少
f. 提供了对称密码技术无法或很难提供的服务：如与哈希函数联合运用可生成数字签名，可证明的安全伪随机数发生器的构造，零知识证明等

7. 加密解密协议（可用于认证）
    a. 保证机密性：用公钥加密，有私钥者才能破译
    b. 保证真实性：用私钥加密，有公钥者都能破译
    c. 既保证机密性又保证真实性：用自己的私钥和对方的公钥加密（签密方案）
        i. 如果先用对方公钥加密，对方可以任意更改公钥来任意伪造收到的信息
8. 公钥密码应满足的要求
    a. 接收方B产生密钥对在计算上是容易的
    b. 发送方A用收方的公钥对消息m加密以产生密文c在计算上是容易的
    c. 收方B用自己的密钥对密文c解密在计算上是容易的
    d. 敌手由密文c和B的公钥恢复明文在计算上是不可行的
    e. 敌手由密文c和B的公钥恢复密钥在计算上是不可行的
    f. 加解密次序可换，即E_PKB[D_SKB(m)]=D_SKB[E_PKB(m)] ,不是对任何算法都做此要求
9. 对公钥密码体制的攻击
    a. 和单钥密码体制一样，如果密钥太短，公钥密码体制也易受到穷搜索攻击。因此密钥必须足够长才能抗击穷搜索攻击。然而又由于公钥密码体制所使用的可逆函数的计算复杂性与密钥长度常常不是呈线性关系，而是增大得更快。所以密钥长度太大又会使得加解密运算太慢而不实用。因此公钥密码体制目前主要用于密钥管理和数字签字
    b. 对公钥密码算法的第2种攻击法是寻找从公开钥计算秘密钥的方法。目前为止，对常用公钥算法还都未能够证明这种攻击是不可行的

-
- 算法

1. RSA
    a. 能够抵抗到目前为止已知的所有密码攻击，已被ISO推荐为公钥数据加密标准
    b. 算法基于大素数因式分解的困难性来产生公钥/私钥对
    c. 应用：银行的u盾、银行卡的刷卡机、淘宝和12306的数字证书
    d. MIT三位年青数学家R.L.Rivest，A.Shamir和L.Adleman [Rivest等1978, 1979]发现了一种用数论构造双钥的方法，称作MIT体制，后来被广泛称之为RSA体制
    e. 它既可用于加密、又可用于数字签字
    f. RSA算法的安全性基于数论中大整数分解的困难性
    g. 迄今为止理论上最为成熟完善的公钥密码体制，该体制已得到广泛的应用
2. RSA密码体制

a. 公钥和私钥的产生：
   i. 随机选择两个大素数p和q，p不等于q，p和q保密
   ii. 计算n=pq，n公开
   iii. 计算欧拉函数$\phi(n)$=(p-1)(q-1)，保密
   iv. 随机选择整数e，1 < e < $\phi(n)$且gcd(e，$\phi(n)$)=1，e公开
   v. 计算d满足：de ≡ 1 (mod $\phi(n)$)，d保密
   vi. (n，e)是公钥，(p, q，d)是私钥
b. 加密变换：对于明文m，密文为 c ≡ m^e mod n
c. 解密变换：对于密文c，明文为 m ≡ c^d = m^ed mod n ≡ m mod n
d.



e. 确定密钥：
   i. 独立地选取两大素数p和q(各100～200位十进制数字)
   ii. 确定n：计算 n=p×q，其欧拉函数值 ϕ(n)=(p - 1)(q - 1)
   iii. 确定e：随机选一整数e，1 <= e < ϕ(n)，gcd(ϕ(n), e)=1
   iv. 确定d：根据ed ≡ 1 mod ϕ(n)在模ϕ(n)下，计算d
   v. 手算d方法：设ed ≡ kϕ(n)+1，则kϕ(n) ≡ e-1 mod e，当ϕ(n)远大于e时，k是形如te+(e-1)的数

3. RSA计算可行性
   a. 产生密钥：
      i. 由于n是公开的，为了避免攻击者用穷举法求出p和q（根据n=pq），应该从足够大的集合中选取p和q，即p和q必须是大素数
      ii. 目前还没有有效的方法可以产生任意大素数，通常使用的方法是：随机挑选一个期望大小的奇数，然后测试它是否是素数，若不是，则挑选下一个随机数直至检测到素数为止
      iii. 定理：如果p为大于2的素数，则方程x^2≡1 mod p的解只有x≡1 mod p 和x≡-1 mod p
      iv. 性质1：p为大于2的素数，如果有x使得x^2≡1 mod p成立，那么x mod p=1或者x mod p=p-1
      v. 性质2：p为大于2的素数，可以表示为p=2^k q+1, k>0, q为一奇数。设a是一个整数(1<a<p−1)，那么下面两个结论必有其一成立：

1) $a^q \mod p=1$;
2) 在$a^q, a^2q, a^4q, \cdots, a^{(2k-1)}q$这些整数中，必有一个数模p所得的余数为p-1

b. 素性检测算法
   i. Find integers k, q, with k > 0, q odd, so that (n-1 = 2kq);
   ii. Select a random integer a, 1 < a < n-1;
   if aq mod n = 1 then return("inconclusive");
   for j = 0 to k 1 do
   if mod n=n-1 then
   return("inconclusive");
   else
   return("composite");
   iii. 算法对s个不同的a，重复调用，如果每次都返回inconclusive，则n是素数的概率大于等于1-2^s
   iv. Miller-Rabin算法可以确定一个整数是合数，但不能确定其一定是素数
   v. 要找到一个2200左右的素数，在找到素数之前大约要进行ln(2200)/2=70次尝试
   vi. 在N附近平均每隔lnN个整数就会有一个素数

c. 确定d和e
   i. 有了p和q，可计算出ϕ(n)=(p - 1)(q - 1)
   ii. 根据gcd(ϕ(n),e)=1来选择e，这一步计算量也不大，因为两个随机数互素的概率约为0.6
   iii. 有了e，再计算d=e-1 mod ϕ(n)，这里用的是扩展的Euclid算法

d. 攻击方法：穷举法或数学分析

e. 安全性：基于分解大整数的困难性假定
   i. 如果分解n=p×q，则立即获得ϕ(n)=(p - 1)(q - 1)，从而能够确定e的模ϕ(n)乘法逆d
   ii. RSA-129历时8个月(曾经预言需要4*1016年)被于1996年4月被成功分解，RSA - 130于1996年4月被成功分解
   iii. 密钥长度应该介于1024bit到2048bit之间
   iv. 由n直接求ϕ(n)等价于分解n

4. 非对称密码算法存在的问题
   a. 速度问题
   b. 中间人攻击：一种常见的攻击方式，攻击者从通信信道中截获数据包并对其进行修改，然后再插回信道中，从而将自己伪装成合法的通信用户
   c.

Protection Lifetime of Data	Present – 2010	Present – 2030	Present – 2031 and Beyond
Minimum symmetric security level	80 bits	112 bits	128 bits
Minimum RSA key size	1024 bits	2048 bits	3072 bits

5. ElGamal密码体制
   a. 由ElGamal于1985年提出
   b. 可用于加解密、数字签名等
   c. 安全性是建立于离散对数问题之上的
      i. 给定g,p与$y=g^x \mod p$，求x是计算上不可行的
      ii. 常取p为1024位
   d. 定义$\mathcal{K}$={$(p,\alpha,A,\beta)|\beta\equiv\alpha^A \pmod p$}，模$p$,本原根$\alpha$,随机$\beta$是公钥，A是私钥
   e. 对于$K=(p,\alpha,A,\beta)$，以及一个（秘密）随机数$k\in\mathbb{Z}_{(p-1)}$，定义：
      i. $e_K(x,k)=(y_1,y_2)$
      ii. 其中$y_1=\alpha^k \mod p$，$y_2=x\beta^k \mod p$

f. 对于$y_1,y_2\in[\mathbb{Z}_p]^*$，定义
   i. $\boldsymbol{d\_K(y\_1,y\_2)=y\_2(y\_1^{A})^{(-1)}\equiv x}\ \beta^k\ (\alpha^{(Ak)})^{-1}\equiv \mathbf{x}\ \mathbf{mod}\ p$
g. 对m用本原根g签名：任选随机数$k\in Z_p$满足gcd(k,$p-1$)=1，任选私钥x$\in Z_p$得公钥y=g^x mod p；求$r=g^k\ mod\ p$，$s=k^{(-1)}(m-x\times r)\ mod\ (p-1)$，则有$y^r\times r^s=g^m\ (mod\ p)$

6. Elgamal密码安全性分析
   a. If adversary can get k from $y\_1=\alpha^k$ mod p, then the scheme is broken.
   b. If adversary can get a from $\beta=\alpha^a$ mod p, then the scheme is broken.
   c. From $y\_1=\alpha^k$ mod p and $\beta=\alpha^a$ mod p, if adversary can compute α^{ka} mod p, then the scheme is broken.
   d. First two correspond to DLP (Discrete Logarithm Problem).
   e. The last one corresponds to Diffie-Hellman Problem.
7. 椭圆曲线密码编码学(Elliptic Curve Cryptography,ECC)
   a. 1985年由Neal Koblitz和Victor Miller分别独立提出的
   b. 160位的椭圆曲线密码算法的安全性相当于1024位的RSA算法
   c. ECC-160加解密的速度比RSA-1024快5-8倍
   d. 椭圆曲线密码作为新的信息安全标准，如IEEE P1364/D4、ANSI F9.62、ANSI F9.63等，分别规范了椭圆曲线密码在Internet协议安全、电子商务、Web服务器、空间通信、移动通信、智能卡等方面的应用
8. 公钥vs对称钥
   a.

Symmetric key	Public key
Two parties MUST trust each other	Two parties DO NOT need to trust each other
Both share the same key (or one key is computable from the other)	Two separate keys: a public key and a private key
Attack approach: bruteforce	Attack approach: solving mathematical problems (e.g. factorization, discrete log problem)
Faster	Slower (100-1000 times slower)
Smaller key size	Larger key size
Examples: DES, Camellia, RC6, AES, ...	Examples: RSA, ElGamal, ECC,...